Smaller models do not exhibit superior generalization performance.

This manuscript (permalink) was automatically generated from greenelab/generalization-manuscript@f3f8850 on July 27, 2023.

Authors

- **Jake Crawford**
  0000-0001-6207-0782 · jjc2718 · jjc2718
  Genomics and Computational Biology Graduate Group, Perelman School of Medicine, University of Pennsylvania, Philadelphia, PA, USA

- **Casey S. Greene** ✉
  0000-0001-8713-9213 · cgreene · GreeneScientist
  Department of Biomedical Informatics, University of Colorado School of Medicine, Aurora, CO, USA; Center for Health AI, University of Colorado School of Medicine, Aurora, CO, USA

✉ — Correspondence possible via GitHub Issues or email to Casey S. Greene <casey.s.greene@cuanschutz.edu>.

Abstract

Introduction

Gene expression datasets are typically “wide”, with many gene features and relatively few samples. These feature-rich datasets present obstacles in many aspects of machine learning, including overfitting and multicollinearity, and challenges in interpretation. To facilitate the use of feature-rich gene expression data in machine learning models, feature selection and/or dimension reduction are commonly used to distill a more condensed data representation from the input space of all genes [1,2]. The intuition is that many gene expression features are likely irrelevant to the prediction problem, redundant, or contain no meaningful variation across samples, so transforming them or selecting a subset can generate a more reliable predictor.

In cancer transcriptomics, this preference for small, parsimonious sets of genes can be seen in the popularity of “gene signatures”. These are groups of genes whose expression levels are used to define cancer subtypes or to predict prognosis or therapeutic response [3,4]. Many studies specify the size of the signature in the paper’s title or abstract, suggesting that the fewer genes in a gene signature, the better, e.g. [5,6,7]. Clinically, there are many reasons why a smaller gene signature may be preferable, including cost (fewer genes may be less expensive to profile or validate, whereas a large signature likely requires a targeted array or NGS analysis [8]) and interpretability (it is easier to reason about the function and biological role of a smaller gene set than a large one since even disjoint gene signatures tend to converge on common biological pathways [9,10]). There is also an underlying assumption that smaller gene signatures tend to be more robust: that for a new patient or in a new biological context, a smaller gene set or more parsimonious model will be more likely to maintain its predictive performance than a larger one. This assumption has rarely been explicitly tested in genomics applications, but is often included in guidelines or rules of thumb for statistical modeling or machine learning in biology, e.g. [11,12,13].

In this study, we sought to test the robustness assumption directly by evaluating model generalization across biological contexts, inspired by previous work on domain adaptation and transfer learning in cancer transcriptomics [14,15,16]. We used two large, heterogeneous public cancer datasets: The Cancer Genome Atlas (TCGA) for human tumor sample data [17], and the Cancer Cell Line Encyclopedia (CCLE) for human cell line data [18]. These datasets contain overlapping -omics data types derived from distinct data sources, allowing us to quantify model generalization across data sources. In addition, each dataset contains samples from a wide range of different cancer types/tissues of origin, allowing us to quantify model generalization across cancer types. We trained both linear and non-linear models to predict mutation status (presence or absence) from RNA-seq gene expression for approximately 70 cancer driver genes, across varying levels of model simplicity and degrees of regularization, resulting in a variety of gene signature sizes. We compared two simple procedures for model selection, one that combines cross-validation performance with model parsimony and one that only relies on cross-validation performance, for each classifier in each context.

Our results suggest that, in general, mutation status classification models that perform well in cross-validation within a biological context also generalize well across biological contexts. There are some individual genes and some individual cancer types where more regularized well-performing models outperform the best-performing model. However, we do not observe a systematic generalization advantage for smaller/more regularized models across all genes and cancer types. These results provide evidence that good cross-validation performance within a biological context (data source or cancer type) is a sufficient proxy for robust performance across contexts.

Methods

Mutation data download and preprocessing

To generate binary mutated/non-mutated gene labels for our machine learning model, we used mutation calls for TCGA samples from MC3 [19] and copy number threshold calls from GISTIC2.0 [20]. MC3 mutation calls were downloaded from the Genomic Data Commons (GDC) of the National Cancer Institute, at https://gdc.cancer.gov/about-data/publications/pancanatlas. Thresholded copy number calls are from an older version of the GDC data and are available here: https://figshare.com/articles/dataset/TCGA_PanCanAtlas_Copy_Number_Data/6144122. We removed hypermutated samples, defined as two or more standard deviations above the mean non-silent somatic mutation count, from our dataset to reduce the number of false positives (i.e., non-driver mutations). Any sample with either a non-silent somatic variant or a copy number variation (copy number gain in the target gene for oncogenes and copy number loss in the target gene for tumor suppressor genes) was included in the positive set; all remaining samples were considered negative for mutation in the target gene.

We followed a similar procedure to generate binary labels for cell lines from CCLE, using the data available on the DepMap download portal at https://depmap.org/portal/download/all/. Mutation information was retrieved from the `OmicsSomaticMutations.csv` data file, and copy number inforamtion was retrieved from the `OmicsCNGene.csv` data file. We thresholded the CNV log-ratios provided by CCLE into binary gain/loss calls using a lower threshold of $\log_2(3/2)$ (i.e. cell lines with a log-ratio below this threshold were considered to have a full copy loss in the corresponding gene), and an upper threshold of $\log_2(5/2)$ (i.e. cell lines with a log-ratio above this threshold were considered to have a full copy gain in the corresponding gene). After applying the same hypermutation criteria that we used for TCGA, no cell lines in CCLE were identified as hypermutated. After preprocesing, 1402 cell lines with mutation and copy number data remained. We then combined non-silent point mutations and copy number gain/loss information into binary labels using the same criteria as for TCGA.

Gene expression data download and preprocessing

RNA sequencing data for TCGA was downloaded from GDC at the same link provided above for the Pan-Cancer Atlas. We discarded non-protein-coding genes and genes that failed to map, and removed tumors that were measured from multiple sites. After filtering to remove hypermutated samples and taking the intersection of samples with both mutation and gene expression data, 9074 TCGA samples remained.

RNA sequencing data for CCLE was downloaded from the DepMap download portal, linked above, in the `CCLE_expression.csv` data file. After taking the intersection of CCLE cell lines with both mutation and gene expression data, 1402 cell lines remained. For experiments making predictions across datasets (i.e., training models on TCGA and evaluating performance on CCLE, or vice-versa) we took the intersection of genes in both datasets, resulting in 16041 gene features. For experiments where only TCGA data was used (i.e., evaluating models on held-out cancer types), we used all 16148 gene features present in TCGA after the filtering described above.

Cancer gene set construction

In order to study mutation status classification for a diverse set of cancer driver genes, we started with the set of 125 frequently altered genes from Vogelstein et al. [21] (all genes from Table S2A). For each target gene, to ensure that the training dataset was reasonably balanced (i.e., that there would

be enough mutated samples to train an effective classifier), we included only cancer types with at least 15 mutated samples and at least 5% mutated samples, which we refer to here as "valid" cancer types. In some cases, this resulted in genes with no valid cancer types, which we dropped from the analysis. Out of the 125 genes originally listed in the Vogelstein et al. cancer gene set, we retained 71 target genes for the TCGA to CCLE analysis, and 70 genes for the CCLE to TCGA analyses. For these analyses, each gene needed at least one valid cancer type in TCGA and one valid cancer type in CCLE, to construct the train and test sets. For the cancer type holdout analysis, we retained 56 target genes: in this case, each gene needed at least two valid cancer types in TCGA to be retained, one to train on and one to hold out.

Classifier setup and cross-validation design

We trained logistic regression classifiers to predict whether or not a given sample had a mutational event in a given target gene using gene expression features as explanatory variables. Our model was trained on gene expression data (X) to predict somatic mutation presence or absence (y) in a target gene. To control for varying mutation burden per sample and to adjust for potential cancer type-specific expression patterns, we included one-hot encoded cancer type and $\log_{10}$(sample mutation count) in the model as covariates. Since gene expression datasets tend to have many dimensions and comparatively few samples, we used a LASSO penalty to perform feature selection [22]. LASSO logistic regression has the advantage of generating sparse models (some or most coefficients are 0), as well as having a single tunable hyperparameter which can be easily interpreted as an indicator of regularization strength/model simplicity.

LASSO ($\l_1$-penalized) logistic regression finds the feature weights $\hat{w} \in \mathbb{R}^p$ solving the following optimization problem:

$$\hat{w} = \text{argmin}_w \left(C \cdot l(X, y; w)\right) + ||w||_1$$

where $i \in \{1, \ldots, n\}$ denotes a sample in the dataset, $X_i \in \mathbb{R}^p$ denotes features (gene expression measurements) from the given sample, $y_i \in \{0, 1\}$ denotes the label (mutation presence/absence) for the given sample, and $l(\cdot)$ denotes the negative log-likelihood of the observed data given a particular choice of feature weights, i.e.

$$l(X, y; w) = -\sum_{i=1}^{n} y_i \log\left(\frac{1}{1 + e^{-w^\top X_i}}\right) + (1 - y_i) \log\left(1 - \frac{1}{1 + e^{-w^\top X_i}}\right)$$

Given weight values $\hat{w}$, it is straightforward to predict the probability of a positive label (mutation in the target gene) $P(y^* = 1 \mid X^*; \hat{w})$ for a test sample X^*:

$$P(y^* = 1 \mid X^*; \hat{w}) = \frac{1}{1 + e^{-\hat{w}^\top X^*}}$$

and the probability of no mutation in the target gene, $P(y^* = 0 \mid X^*; \hat{w})$, is given by (1 - the above quantity).

This optimization problem leaves one hyperparameter to select: C, which controls the inverse of the strength of the L1 penalty on the weight values (i.e. regularization strength scales with $\frac{1}{C}$). Although the LASSO optimization problem does not have a closed form solution, the loss function is convex, and iterative optimization algorithms are commonly used for finding reasonable solutions. For fixed values of C, we solved for $\hat{w}$ using `scikit-learn`'s `LogisticRegression` method [23], which

uses the coordinate descent optimization method implemented in `liblinear` [24]. We selected this implementation rather than the `SGDClassifier` stochastic gradient descent implementation because coordinate descent/ `liblinear` tends to generate sparser models and does not depend on a learning rate parameter, although after hyperparameter tuning performance is generally comparable between the implementations [25].

To assess model selection across contexts (datasets and cancer types), we trained models using a variety of LASSO parameters on 75% of the training dataset, holding out 25% of the training dataset as the "cross-validation" set and also evaluating across contexts as the "test" set. We trained models using C values evenly spaced on a logarithmic scale between (10^{-3}, 10^{7}); i.e. the output of `numpy.logspace(-3, 7, 21)`. This range was intended to give evenly distributed coverage across genes and cancer types that included "underfit" models (predicting only the mean or using very few features, poor performance on all datasets), "overfit" models (performing perfectly on training data but comparatively poorly on cross-validation and test data), and a wide variety of models in between that typically included the best fits to the cross-validation and test data. To assess variability between train/CV splits, we used all 4 splits (25% holdout sets) x 2 random seeds for a total of 8 different training sets for each gene, using the same test set (i.e. all of the held-out context, either one cancer type or one dataset) in each case.

"Best model" vs. "smallest good model" analysis

For the "best" vs. "smallest good" model selection comparison, we started with 8 performance measurements (4 cross-validation folds x 2 random seeds) for each of 21 LASSO parameters. We took the mean over these 8 measurements to get a single performance measurement for each model (LASSO parameter) on the holdout dataset, which has the same composition as the training set. We used these per-parameter mean performance measurements to select the "best" model (LASSO parameter with the best performance on the holdout dataset), and the "smallest good" model (smallest LASSO parameter with performance in the top 25% of mean values on the holdout dataset, rounded up to the nearest integer). For the distributions of differences shown in the Results, we took the difference in mean performance for the "best" and "smallest good" models for each gene, with positive differences indicating better performance for the "best" model and negative differences better performance for the "smallest good" model, for each gene.

Neural network setup and parameter selection

Inspired by the intermediate-complexity model in [26], as a tradeoff between computational cost and ability to represent non-linear decision boundaries, we trained a three-layer fully connected neural network with ReLU nonlinearities [27] to predict mutation status. For the experiments described in the main paper, we varied the size of the first hidden layer in the range {1, 2, 3, 4, 5, 10, 50, 100, 500, 1000}. We fixed the size of the second hidden layer to be half of the size of the first hidden layer, rounded up to the nearest integer, and the size of the third hidden layer was the number of classes, 2 in our case. Our models were trained for 100 epochs of mini-batch stochastic gradient descent in PyTorch [28], using the Adam optimizer [29] and a fixed batch size of 50. To select the remaining hyperparameters for each hidden layer size, we performed a random search over 10 combinations, with a single train/test split stratified by cancer type, using the following hyperparameter ranges: learning rate {0.1, 0.01, 0.001, 5e-4, 1e-4}, dropout proportion {0.1, 0.5, 0.75}, weight decay (L2 penalty) {0, 0.1, 1, 10, 100}. We used the same train/cross-validation split strategy described above, generating 8 different performance measurements for each gene and hidden layer size, for the neural network experiments as well.

For the *EGFR* gene, we also ran experiments where we varied the dropout proportion and the weight decay hyperparameter as the regularization axis, and selected the remaining hyperparameters

(including the hidden layer size) using a random search. In these cases, we used a fixed range for dropout of {0.0, 0.05, 0.125, 0.25, 0.375, 0.5, 0.625, 0.75, 0.875, 0.95}, and a fixed range for weight decay of {0.0, 0.001, 0.005, 0.01, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.75, 1.0, 10.0}. All neural network analyses were performed on a Ubuntu 18.04 machine with a NVIDIA RTX 2060 GPU.

Results

Evaluating model generalization using public cancer data

We collected data from the TCGA Pan-Cancer Atlas and the Cancer Cell Line Encyclopedia to predict the presence or absence of mutations in cancer genes, as a benchmark of cancer-related information content across cancer types and contexts. We trained mutation status classifiers across approximately 70 genes involved in cancer development and progression from Vogelstein et al. 2013 [30], using LASSO logistic regression with gene expression (RNA-seq) values as predictive features. We fit each classifier across a variety of regularization parameters, resulting in models with a variety of different sparsity levels between the extremes of 0 nonzero features and all features included (Supplementary Figure S1). Inspired by the generalization experiments across tissues and model systems in [14], we designed experiments to evaluate the generalization of mutation status classifiers across datasets (TCGA to CCLE and CCLE to TCGA) and across biological contexts (cancer types) within TCGA, relative to a within-dataset baseline (Figure 1).

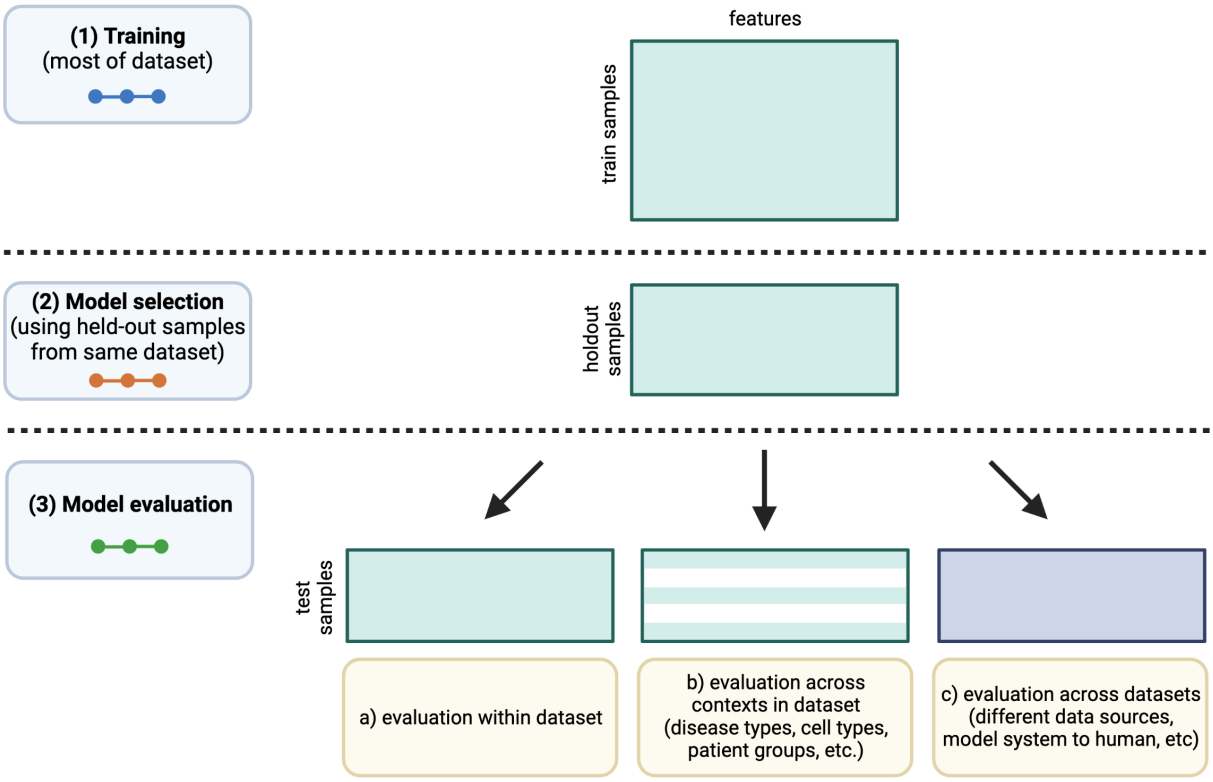

Figure 1: Schematic of experimental design. The colors of the "dots" in the training/model selection/model evaluation panels on the left correspond to train/CV/test curves in the following results figures.

Generalization from human tumor samples to cell lines is more effective than the reverse

To evaluate "cross-dataset" generalization, we trained mutation status classifiers on human tumor data from TCGA and evaluated them on cell line data from CCLE, as well as the reverse from CCLE to TCGA. As an example, we examined *EGFR*, an oncogenic tyrosine kinase that is commonly mutated in diverse cancer types and cancer cell lines, including lung cancer, colorectal cancer, and glioblastoma [31,32]. For EGFR mutation status classifiers trained on TCGA and evaluated on CCLE, we saw that AUPR on cell lines was slightly worse than on held-out tumor samples, but comparable across regularization levels/LASSO parameters (Figure 2A). On the other hand, EGFR classifiers trained on

CCLE and evaluated on TCGA performed considerably worse on human tumor samples as compared to held-out cell lines (Figure 2B).

To explore these tendencies more generally, we compared performance across all genes in the Vogelstein et al. dataset, for both TCGA to CCLE and CCLE to TCGA generalization. We measured the difference between performance on the holdout data within the training dataset and performance across datasets, with a positive difference indicating poor generalization (better holdout performance than test performance) and a 0 or negative difference indicating good generalization (comparable test performance to holdout performance). For generalization from TCGA to CCLE, we observed that median AUPR differences were mostly centered around 0 for most genes, with some exceptions at the extremes (Figure 2C; performance differences on the y-axis). An example of a gene exhibiting poor generalization was *IDH1*, the leftmost gene in Figure 2C, with good performance on held-out TCGA data and poor performance on CCLE data. IDH-mutant glioma cell lines are poorly represented compared to IDH-mutant patient tumors, which may explain the difficulty of generalization to cell lines for *IDH1* mutation classifiers [33]. For generalization from CCLE to TCGA, we observed a more pronounced upward shift toward better performance on CCLE and worse on TCGA, with most genes performing better on the CCLE holdout data and very few genes generalizing comparably to the TCGA samples (Figure 2D).

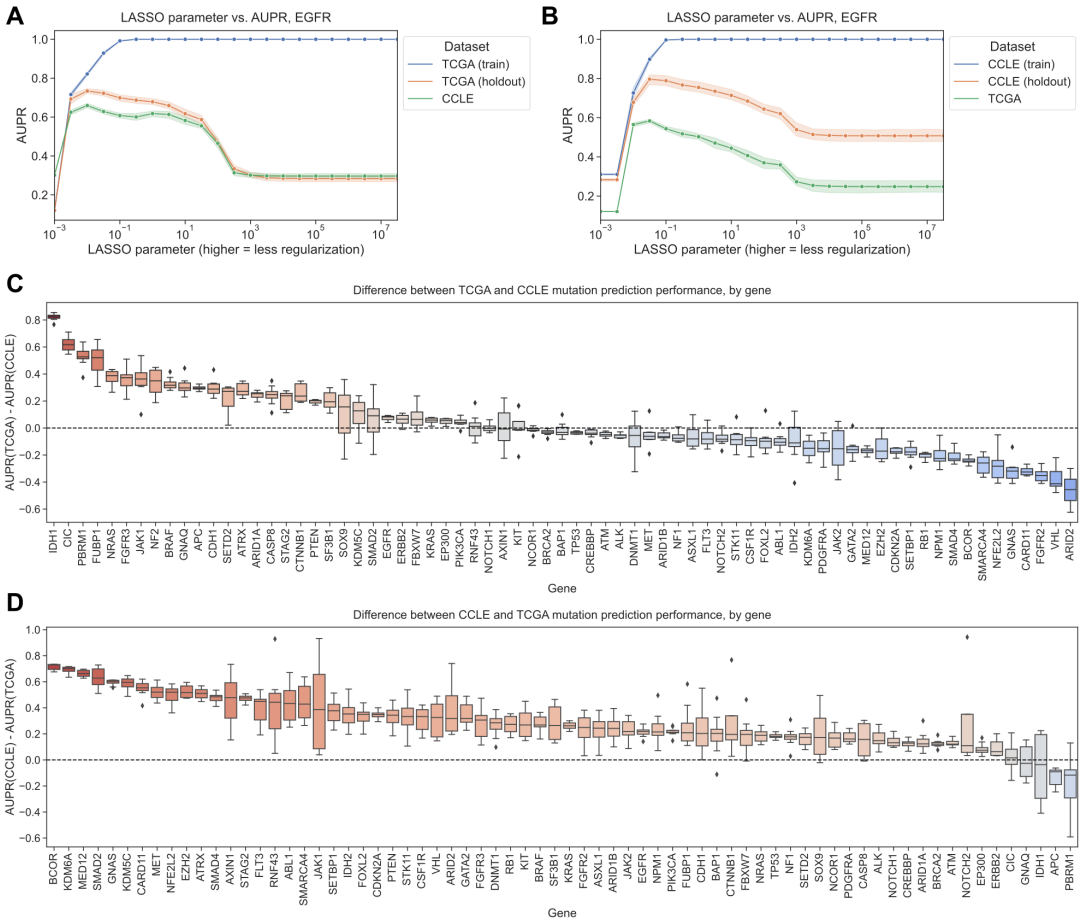

Figure 2: **A.** *EGFR* mutation status prediction performance on training samples from TCGA (blue), held-out TCGA samples (orange), and CCLE samples (green), across varying LASSO parameters. **B.** *EGFR* mutation status prediction performance on training samples from CCLE (blue), held-out CCLE samples (orange), and TCGA samples (green). **C.** Difference in mutation status prediction performance for models trained on TCGA (holdout data) and evaluated on CCLE (test data), across 71 genes from Vogelstein et al. For each gene, the best model (LASSO parameter) was selected using holdout AUPR performance. Genes on x-axis are ordered by median AUPR difference across cross-validation splits, from highest to lowest. **D.** Difference in mutation status prediction performance for models trained on CCLE (holdout data) and evaluated on TCGA (test data), across 70 genes from Vogelstein et al.

“Best” and “smallest good” model selection strategies perform comparably

To address the question of whether more parsimonious models tend to generalize better or not, we designed two model selection schemes and compared them for the TCGA to CCLE and CCLE to TCGA mutation prediction problems (Figure 3A). The “best” model selection scheme chooses the top-performing model/LASSO parameter on the holdout dataset from the same source as the training data and applies it to the test data from the other data source. The intention of the “smallest good” model selection scheme is to balance parsimony with reasonable performance on the holdout data, since simply selecting the smallest possible model (generally, the dummy regressor/mean predictor) is not likely to generalize well. To accomplish this, we first identify the top 25% of well-performing models on the holdout dataset; then, from this subset of models, we choose the smallest (i.e., highest LASSO parameter) to apply to the test data. In both cases, we exclusively use the holdout data to select a model and only apply the model to out-of-dataset samples to evaluate generalization performance *after* model selection.

For TCGA to CCLE generalization, 31/71 genes (43.7%) had better performance for the “best” model, and 15/71 genes (21.1%) had better generalization performance with the “smallest good” model. The other 25 genes had the same “best” and “smallest good” model (in other words, the “smallest good” model was also the best-performing overall, and the difference was 0) (Figure 3B). For CCLE to TCGA generalization, 24/70 genes (34.3%) had better performance for the “best” model and 19/70 (27.1%) for the “smallest good,” with the other 27 having the same model fulfill both criteria (Figure 3C). Overall, these results do not support the hypothesis that the most parsimonious model generalizes the best: for both generalization problems there are slightly more genes where the best-performing model on the holdout dataset is also the best-performing on the test set, although there are some genes where the “smallest good” approach works well.

We examined genes that fell into either category for TCGA to CCLE generalization (dotted lines on Figure 3B). For *NF1*, the “best” model outperforms the “smallest good” model (Figure 3D). Comparing holdout (orange) and cross-dataset (green) performance, both generally follow a similar trend, with the cross-dataset performance peaking when the holdout performance peaks at a regularization parameter of $\alpha = 0.00316$. *PIK3CA* is an example of the opposite, a gene where the “smallest good” model tends to outperform the “best” model (Figure 3E). In this case, the peak for the cross-dataset performance occurs at a higher level of regularization (further left on the x-axis), at $\alpha = 0.01$, than the peak for the holdout performance, at $\alpha = 0.0316$. This suggests that a *PIK3CA* mutation status classifier that is more parsimonious, but that has slightly worse performance, does tend to generalize better across datasets to CCLE.

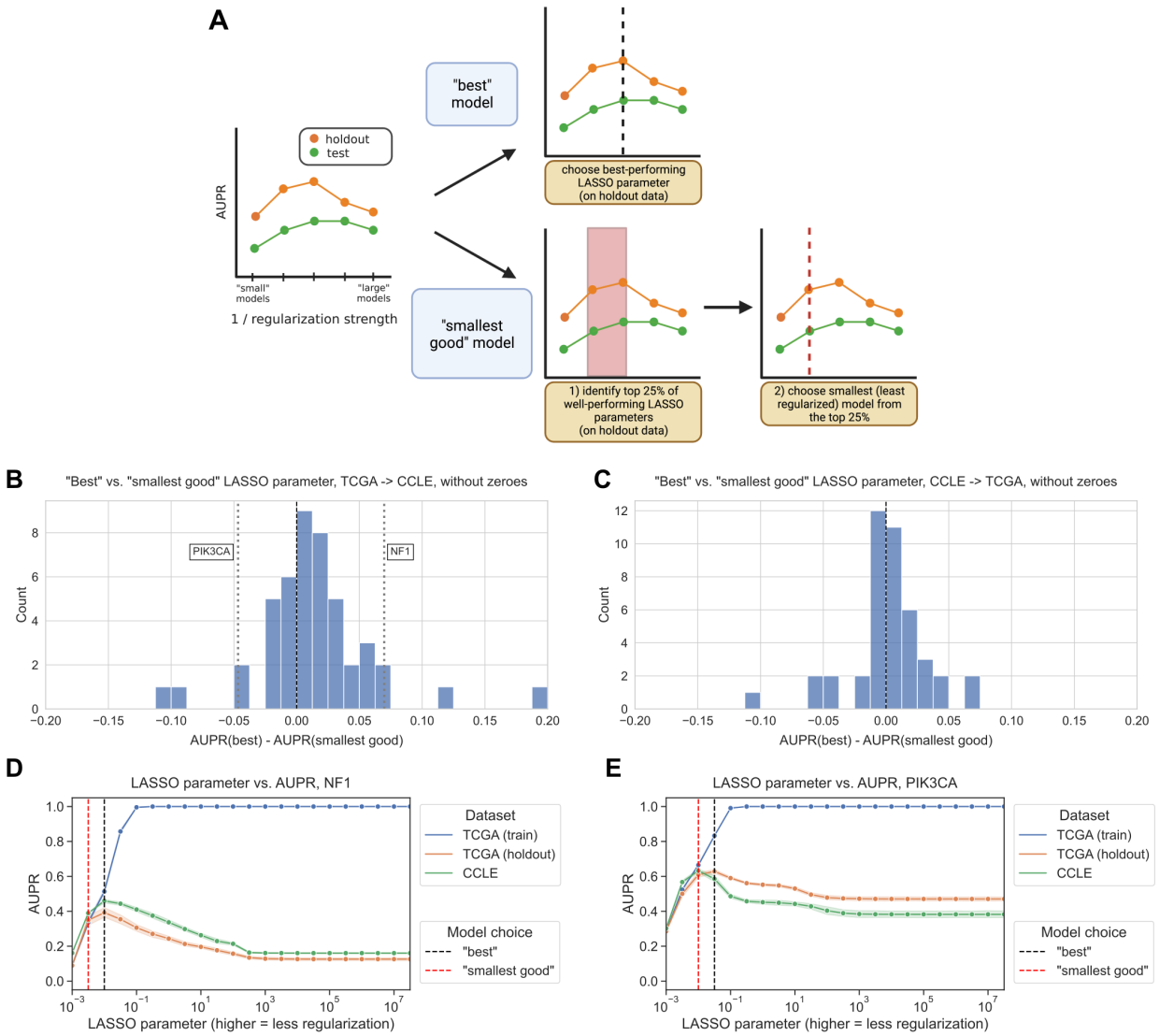

Figure 3: **A.** Schematic of "best" vs. "smallest good" model comparison experiments. **B.** Distribution of performance comparisons between "best" and "smallest good" model selection strategies, for TCGA -> CCLE generalization. Positive x-axis values indicate better performance for the "best" model, negative values indicate better performance for the "smallest good" model. **C.** Distribution of performance comparisons between "best" and "smallest good" model selection strategies, for CCLE -> TCGA generalization. **D.** *NF1* mutation status prediction performance generalizing from TCGA (holdout, orange), to CCLE (green), with "best" and "smallest good" models labeled. **E.** *PIK3CA* mutation status prediction performance generalizing from TCGA (holdout, orange), to CCLE (green), with "best" and "smallest good" models labeled.

Generalization across cancer types yields similar results to generalization across datasets

To evaluate generalization across biological contexts within a dataset, we trained mutation prediction classifiers on all but one cancer type in TCGA, performed model selection on a holdout set stratified by cancer type, and held out the remaining cancer type as a test set. We performed the same "best" vs. "smallest good" analysis that was previously described, across 294 gene/holdout cancer type combinations (Figure 4A). We observed 133/294 gene/cancer type combinations (45.2%) that had better generalization performance with the "best" model, compared to 84/294 (28.6%) for the "smallest good" model. The other 77 gene/cancer type combinations had the same "best" and "smallest good" model and thus no difference in performance. This is consistent with our cross-dataset experiments, with slightly more instances where the "best" model on the stratified holdout data also generalizes the best, but no pronounced distributional shift in either direction.

We looked in more detail at two examples of gene/cancer type combinations, one on either side of the 0 point for cross-cancer type generalization. For prediction of *SETD2* mutation status in papillary renal

cell carcinoma, we observed the best cross-cancer type performance for relatively low levels of regularization/high x-axis values (Figure 4B). For prediction of *CDKN2A* mutation status in low grade glioma, on the other hand, we observed the best cross-cancer generalization for a high level of regularization ($\alpha = 0.01$), and generalization capability for the best parameter on the stratified holdout set ($\alpha = 0.0316$) was lower (Figure 4C).

We aggregated results across genes for each cancer type, looking at performance in the held-out cancer type compared to performance on the stratified holdout set (Figure 4D). Cancer types that were particularly difficult to generalize to (better performance on stratified data than cancer type holdout, or positive y-axis values) include testicular cancer (TGCT) and soft tissue sarcoma (SARC), which are notable because they are not carcinomas like the majority of cancer types included in TCGA, potentially making generalization harder. We also aggregated results across cancer types for each gene, identifying a distinct set of genes where classifiers tend to generalize poorly no matter what cancer type is held out (Supplementary Figure S2). Included in this set of genes with poor generalization performance are *HRAS*, *NRAS*, and *BRAF*, suggesting that a classifier that combines mutations in Ras pathway genes into a single "pathway mutation status" label (as described in [34]) could be a better approach than separate classifiers for each gene.

In the cancer type aggregation plot (Figure 4D), thyroid carcinoma (THCA) stood out as a carcinoma that had poor performance when held out. In our experiments, the only genes in which THCA is included as a held-out cancer type are *BRAF* and *NRAS*; generalization performance for both genes is below cross-validation performance, but slightly worse for *NRAS* than *BRAF* (Supplementary Figure S3). Previous work suggests that *BRAF* mutation tends to have a different functional signature in THCA than other cancer types, and withholding THCA from the training set improved classifier performance, which could at least in part explain the difficulty of generalizing to THCA we observe [34].

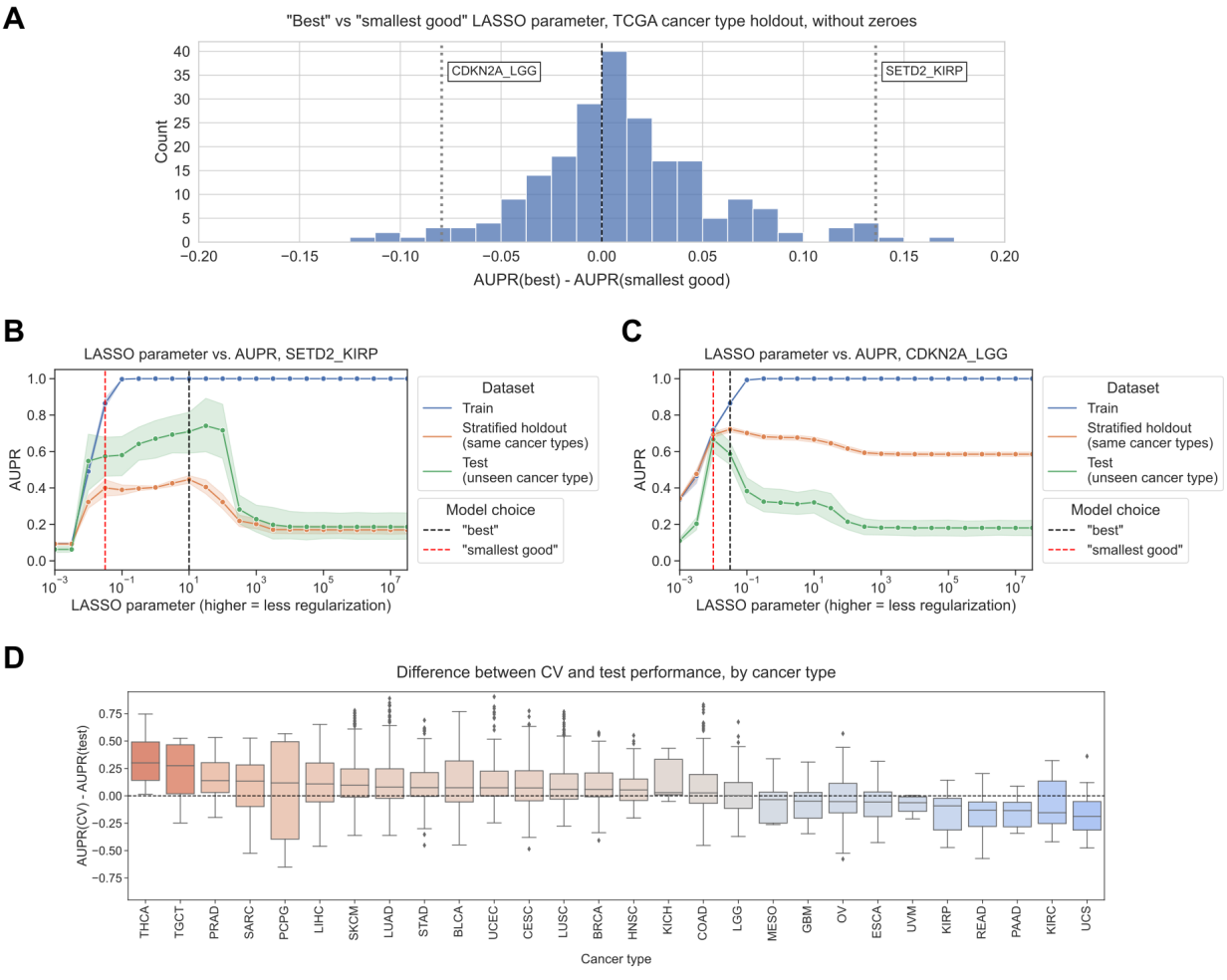

Figure 4: **A.** Distribution of performance comparisons between "best" and "smallest good" model selection strategies, for generalization across TCGA cancer types. Each point is a gene/cancer type combination; positive x-axis values indicate better performance for the "best" model and negative values indicate better performance for the "smallest good" model. **B.** *SETD2* mutation status prediction performance generalizing from other cancer types in TCGA (stratified holdout, orange) to papillary renal cell carcinoma (KIRP, green), with "best" and "smallest good" models labeled. **C.** *CDKN2A* mutation status prediction performance generalizing from other cancer types in TCGA (stratified holdout, orange) to low grade glioma (LGG, green), with "best" and "smallest good" models labeled. **D.** Distributions of performance difference between CV data (same cancer types as train data) and holdout data (cancer types not represented in train data), by held-out cancer type. Each point is a gene whose mutation status classifier was used to make predictions on out-of-dataset samples in the relevant cancer type.

Small neural network hidden layer sizes tend to generalize poorly

To test whether or not findings generalize to non-linear models, we trained a 3-layer neural network to predict mutation status from gene expression for generalization from TCGA to CCLE, and we varied the size of the first hidden layer to control regularization/model complexity. We fixed the size of the second hidden layer to be half the size of the first layer, rounded up to the nearest integer; further details in Methods. For *EGFR* mutation status prediction, we saw that performance for small hidden layer sizes was noisy, but generally lower than for higher hidden layer sizes (Figure 5A). On average, over all 71 genes from Vogelstein et al., performance on both held-out TCGA data and CCLE data tends to increase until a hidden layer size of 10-50, then flatten (Figure 5B). To explore additional approaches to neural network regularization, we also tried varying dropout and weight decay for *EGFR* and *KRAS* mutation status classification while holding the hidden layer size constant. Results followed a similar trend, with generalization performance generally tracking performance on holdout data (Supplementary Figure S4).

In order to measure which hidden layer sizes tended to perform relatively well or poorly, across different mutated cancer genes with different effect sizes, we ranked the range of hidden layer sizes by their generalization performance on CCLE (with low ranks representing good performance, and high ranks representing poor performance; Figure 5C). For each hidden layer size, we then visualized the distribution of ranks above and below the median rank of 5.5/10; a high proportion of ranks above the median (True, or blue bar) signifies poor overall performance for that hidden layer size, and a high proportion of ranks below the median (False, or orange bar) signifies good performance. We saw that small hidden layer sizes tended to generalize poorly (<5, but most pronounced for 1 and 2), and intermediate hidden layer sizes tended to generalize well (10-100, and sometimes 500/1000). This suggests that some degree of parsimony/simplicity could be useful, but very simple models do not tend to generalize well. We also performed the same "best"/"smallest good" analysis as with the linear models, using hidden layer size as the regularization axis instead of LASSO regularization strength. We observed a distribution centered around 0, suggesting that the "best" and "smallest good" models tend to generalize similarly (Figure 5D). 28/71 genes (45.2%) had better generalization performance with the "best" model, compared to 21/71 (28.6%) for the "smallest good" model and 22 with the same "best" and "smallest good" model.

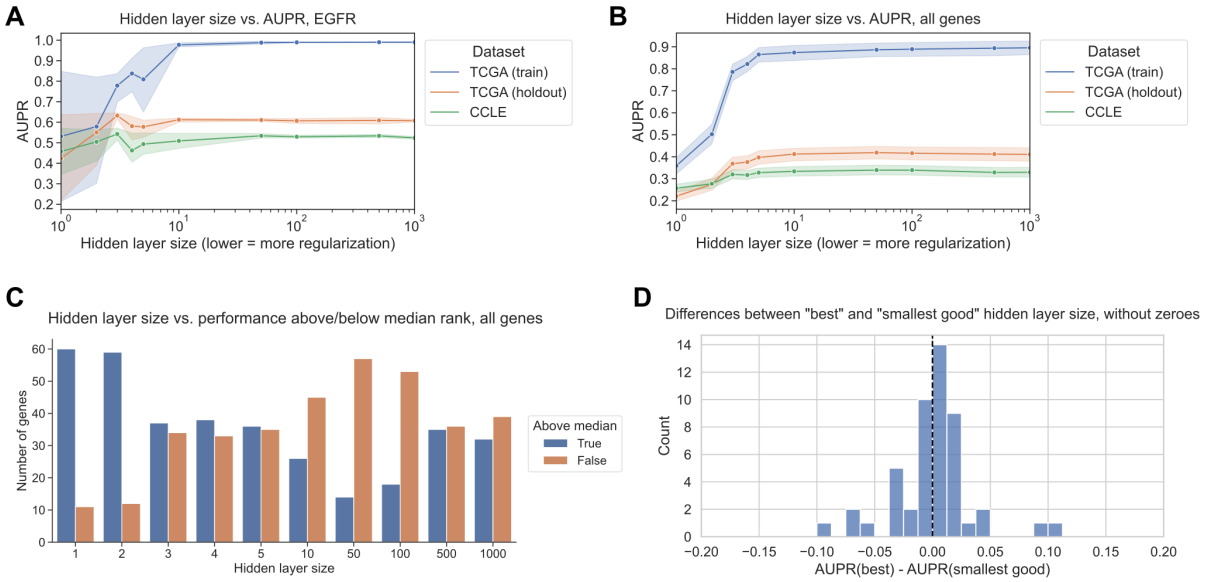

Figure 5: **A.** *EGFR* mutation status prediction performance on training samples from TCGA (blue), held-out TCGA samples (orange), and CCLE samples (green), across varying neural network hidden layer sizes. **B.** Mutation status prediction performance summarized across all genes from Vogelstein et al. on training samples from TCGA (blue), held-out TCGA samples (orange), and CCLE samples (green), across varying neural network hidden layer sizes. **C.** Distribution of ranked performance values above/below the median rank for each gene, for each of the hidden layer sizes evaluated. Lower ranks indicate better performance and higher ranks indicate worse performance, relative to other hidden layer sizes. **D.** Distribution of performance comparisons between "best" and "smallest good" model selection strategies, for TCGA -> CCLE generalization with neural network hidden layer size as the regularization axis. Positive x-axis values indicate better performance for the "best" model, negative values indicate better performance for the "smallest good" model.

Discussion

Using public cancer genomics and transcriptomics data from TCGA and CCLE, we studied generalization of mutation status classifiers for a wide variety of cancer driver genes. We designed experiments to evaluate generalization across biological contexts by holding out cancer types in TCGA, and to evaluate generalization across datasets by training models on TCGA and evaluating them on CCLE, and vice-versa. We found that, in general, smaller or more parsimonious models do not tend to generalize more effectively across cancer types or across datasets, and in the absence of prior knowledge about a prediction problem, simply choosing the model that performs the best on a holdout dataset is at least as effective for selecting models that generalize.

Our results were similar in both linear models (LASSO logistic regression) and non-linear deep neural networks when using hidden layer size as the regularization parameter of interest. In our non-linear model experiments, we did not observe better generalization across datasets for fully connected neural networks with fewer hidden layer nodes, and our preliminary results indicated a similar trend for dropout and weight decay. Compared to linear models, it is less clear how to define a "small" or "parsimonious" neural network model since there are many regularization techniques that one may use to control complexity. Rather than simply removing nodes and keeping the network fully connected, another approach to parsimony could be to select an inductive bias to guide the size reduction of the network. Existing examples include network structures guided by protein-protein interaction networks or function/pathway ontologies [35,36,37,38]. It is possible that a smaller neural network with a structure that corresponds more appropriately to the prediction problem would achieve better generalization results, although choosing an apt network structure or data source can be a challenging aspect of such efforts.

For generalization from CCLE to TCGA, we observed that performance was generally worse on human tumor samples from TCGA than for held-out cell lines. This could, at least in part, be a function of sample size: the number of cell lines in CCLE is approximately an order of magnitude smaller than the number of tumor samples in TCGA (~10,000 samples in TCGA vs. ~1,500 cell lines in CCLE, although the exact number of samples used to train and evaluate our classifiers varies by gene, see Methods for further detail). There are also plausible biological and technical explanations for the difficulty of generalizing to human tumor samples. This result could reflect the imperfect and limited nature of cancer cell lines as a model system for human tumors, which previous studies have pointed out [39,40,41]. In addition, the CCLE data is collected and processed uniformly, as described in [18], while the TCGA data is processed by a uniform pipeline but collected from a wide variety of different cancer centers around the US [17].

When we ranked cancer types in order of their generalization difficulty aggregated across genes, we noticed a slight tendency toward non-carcinoma cancer types (TGCT, SARC, SKCM) being difficult to generalize to. It has been pointed out in other biological data types that holding out entire contexts or domains is necesssary for a full picture of performance [42,43], which our results corroborate. This highlights a potential weakness of using TCGA's carcinoma-dominant pan-cancer data as a training set for a broad range of tasks, for instance in foundation models which are becoming feasible for some genomics applications [44,45,46]. One caveat of our analysis is that each cancer type is included in the training data or held out for a different subset of genes, so it is difficult to detangle gene-specific effects (some mutations have less distinguishable functional effects on gene expression than others) from cancer type-specific effects (some cancer types are less similar to each other than others) on prediction performance using our experimental design.

Data and code availability

The data from TCGA analyzed during this study were previously published as part of the TCGA Pan-Cancer Atlas project [17], and are available from the NIH NCI Genomic Data Commons (GDC). The data from CCLE analyzed during this study were previously published [18], and are available from the Broad Institute's DepMap Portal. The scripts used to download and preprocess the datasets for this study are available at https://github.com/greenelab/pancancer-evaluation/tree/master/00_process_data. Scripts for TCGA <-> CCLE comparisons (Figures 2 and 3) and neural network experiments (Figure 5) are available in the https://github.com/greenelab/pancancer-evaluation/tree/master/08_cell_line_prediction directory. Scripts for TCGA cancer type comparisons (Figure 4) are available in the https://github.com/greenelab/pancancer-evaluation/tree/master/02_cancer_type_classification directory. All scripts are available under the open-source BSD 3-clause license.

This manuscript was written using Manubot [47] and is available on GitHub at https://github.com/greenelab/generalization-manuscript under the CC0-1.0 license. This research was supported in part by the University of Pittsburgh Center for Research Computing through the resources provided. Specifically, this work used the HTC cluster, which is supported by NIH award number S10OD028483.

References

1. **Effective dimension reduction methods for tumor classification using gene expression data**
A Antoniadis, S Lambert-Lacroix, F Leblanc
Bioinformatics (2003-03-22) https://doi.org/dhfzst
DOI: 10.1093/bioinformatics/btg062 · PMID: 12651713

2. **Feature selection and dimension reduction for single-cell RNA-Seq based on a multinomial model**
FWilliam Townes, Stephanie C Hicks, Martin J Aryee, Rafael A Irizarry
Genome Biology (2019-12) https://doi.org/ggk85t
DOI: 10.1186/s13059-019-1861-6 · PMID: 31870412 · PMCID: PMC6927135

3. **Cancer transcriptome profiling at the juncture of clinical translation**
Marcin Cieślik, Arul M Chinnaiyan
Nature Reviews Genetics (2017-12-27) https://doi.org/gcsmnr
DOI: 10.1038/nrg.2017.96 · PMID: 29279605

4. **Cancer gene expression signatures – The rise and fall?**
Frederic Chibon
European Journal of Cancer (2013-05) https://doi.org/f2gtqf
DOI: 10.1016/j.ejca.2013.02.021 · PMID: 23498875

5. **A Five-Gene Signature and Clinical Outcome in Non–Small-Cell Lung Cancer**
Hsuan-Yu Chen, Sung-Liang Yu, Chun-Houh Chen, Gee-Chen Chang, Chih-Yi Chen, Ang Yuan, Chiou-Ling Cheng, Chien-Hsun Wang, Harn-Jing Terng, Shu-Fang Kao, … Pan-Chyr Yang
New England Journal of Medicine (2007-01-04) https://doi.org/dsnktr
DOI: 10.1056/nejmoa060096 · PMID: 17202451

6. **A Six-Gene Signature Predicting Breast Cancer Lung Metastasis**
Thomas Landemaine, Amanda Jackson, Akeila Bellahcène, Nadia Rucci, Soraya Sin, Berta Martin Abad, Angels Sierra, Alain Boudinet, Jean-Marc Guinebretière, Enrico Ricevuto, … Keltouma Driouch
Cancer Research (2008-08-01) https://doi.org/frmj5f
DOI: 10.1158/0008-5472.can-08-0436 · PMID: 18676831

7. **70-Gene Signature as an Aid to Treatment Decisions in Early-Stage Breast Cancer**
Fatima Cardoso, Laura J van't Veer, Jan Bogaerts, Leen Slaets, Giuseppe Viale, Suzette Delaloge, Jean-Yves Pierga, Etienne Brain, Sylvain Causeret, Mauro DeLorenzi, … Martine Piccart
New England Journal of Medicine (2016-08-25) https://doi.org/gdp988
DOI: 10.1056/nejmoa1602253 · PMID: 27557300

8. **MammaPrint™ 70-gene signature: another milestone in personalized medical care for breast cancer patients**
Elzbieta A Slodkowska, Jeffrey S Ross
Expert Review of Molecular Diagnostics (2009-07) https://doi.org/c8qptt
DOI: 10.1586/erm.09.32 · PMID: 19580427

9. **Sorting Out Breast-Cancer Gene Signatures**
Joan Massagué
New England Journal of Medicine (2007-01-18) https://doi.org/cbt3x7
DOI: 10.1056/nejme068292 · PMID: 17229957

10. **Challenges translating breast cancer gene signatures into the clinic**
Britta Weigelt, Lajos Pusztai, Alan Ashworth, Jorge S Reis-Filho
Nature Reviews Clinical Oncology (2011-08-30) https://doi.org/cp83ks
DOI: 10.1038/nrclinonc.2011.125 · PMID: 21878891

11. **What do we mean by validating a prognostic model?**
Douglas G Altman, Patrick Royston
Statistics in Medicine (2000-02-29) https://doi.org/bhfhgd
DOI: 10.1002/(sici)1097-0258(20000229)19:4<453::aid-sim350>3.0.co;2-5

12. **Evaluating Microarray-based Classifiers: An Overview**
A-L Boulesteix, C Strobl, T Augustin, M Daumer
Cancer Informatics (2008-01) https://doi.org/ggsmz4
DOI: 10.4137/cin.s408 · PMID: 19259405 · PMCID: PMC2623308

13. **Ten Simple Rules for Effective Statistical Practice**
Robert E Kass, Brian S Caffo, Marie Davidian, Xiao-Li Meng, Bin Yu, Nancy Reid
PLOS Computational Biology (2016-06-09) https://doi.org/gcx4rn
DOI: 10.1371/journal.pcbi.1004961 · PMID: 27281180 · PMCID: PMC4900655

14. **Few-shot learning creates predictive models of drug response that translate from high-throughput screens to individual patients**
Jianzhu Ma, Samson H Fong, Yunan Luo, Christopher J Bakkenist, John Paul Shen, Soufiane Mourragui, Lodewyk FA Wessels, Marc Hafner, Roded Sharan, Jian Peng, Trey Ideker
Nature Cancer (2021-01-25) https://doi.org/gh52nt
DOI: 10.1038/s43018-020-00169-2 · PMID: 34223192 · PMCID: PMC8248912

15. **Out-of-distribution generalization from labelled and unlabelled gene expression data for drug response prediction**
Hossein Sharifi-Noghabi, Parsa Alamzadeh Harjandi, Olga Zolotareva, Colin C Collins, Martin Ester
Nature Machine Intelligence (2021-11-11) https://doi.org/gq32k7
DOI: 10.1038/s42256-021-00408-w

16. **Predicting patient response with models trained on cell lines and patient-derived xenografts by nonlinear transfer learning**
Soufiane MC Mourragui, Marco Loog, Daniel J Vis, Kat Moore, Anna G Manjon, Mark A van de Wiel, Marcel JT Reinders, Lodewyk FA Wessels
Proceedings of the National Academy of Sciences (2021-12-03) https://doi.org/gshpgt
DOI: 10.1073/pnas.2106682118 · PMID: 34873056 · PMCID: PMC8670522

17. **The Cancer Genome Atlas Pan-Cancer analysis project**
John N Weinstein, Eric A Collisson, Gordon B Mills, Kenna RMills Shaw, Brad A Ozenberger, Kyle Ellrott, Ilya Shmulevich, Chris Sander, Joshua M Stuart
Nature Genetics (2013-09-26) https://doi.org/f3nt5c
DOI: 10.1038/ng.2764 · PMID: 24071849 · PMCID: PMC3919969

18. **Next-generation characterization of the Cancer Cell Line Encyclopedia**
Mahmoud Ghandi, Franklin W Huang, Judit Jané-Valbuena, Gregory V Kryukov, Christopher C Lo, ERobert McDonald III, Jordi Barretina, Ellen T Gelfand, Craig M Bielski, Haoxin Li, … William R Sellers
Nature (2019-05) https://doi.org/gf2m3h
DOI: 10.1038/s41586-019-1186-3 · PMID: 31068700 · PMCID: PMC6697103

19. **Scalable Open Science Approach for Mutation Calling of Tumor Exomes Using Multiple Genomic Pipelines**
Kyle Ellrott, Matthew H Bailey, Gordon Saksena, Kyle R Covington, Cyriac Kandoth, Chip Stewart, Julian Hess, Singer Ma, Kami E Chiotti, Michael McLellan, … Armaz Mariamidze
Cell Systems (2018-03) https://doi.org/gf9twn
DOI: 10.1016/j.cels.2018.03.002 · PMID: 29596782 · PMCID: PMC6075717

20. **GISTIC2.0 facilitates sensitive and confident localization of the targets of focal somatic copy-number alteration in human cancers**
Craig H Mermel, Steven E Schumacher, Barbara Hill, Matthew L Meyerson, Rameen Beroukhim, Gad Getz
Genome Biology (2011-04) https://doi.org/dzhjqh
DOI: 10.1186/gb-2011-12-4-r41 · PMID: 21527027 · PMCID: PMC3218867

21. **Evaluating the evaluation of cancer driver genes**
Collin J Tokheim, Nickolas Papadopoulos, Kenneth W Kinzler, Bert Vogelstein, Rachel Karchin
Proceedings of the National Academy of Sciences (2016-11-22) https://doi.org/f9d77w
DOI: 10.1073/pnas.1616440113 · PMID: 27911828 · PMCID: PMC5167163

22. **Regression Shrinkage and Selection Via the Lasso**
Robert Tibshirani
Journal of the Royal Statistical Society: Series B (Methodological) (1996-01)
https://doi.org/gfn45m
DOI: 10.1111/j.2517-6161.1996.tb02080.x

23. **Scikit-learn: Machine Learning in Python**
Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, … Édouard Duchesnay
Journal of Machine Learning Research (2011) http://jmlr.org/papers/v12/pedregosa11a.html

24. **LIBLINEAR: A Library for Large Linear Classification**
Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, Chih-Jen Lin
Journal of Machine Learning Research (2008) http://jmlr.org/papers/v9/fan08a.html

25. **Optimizer's dilemma: optimization strongly influences model selection in transcriptomic prediction**
Jake Crawford, Maria Chikina, Casey S Greene
Cold Spring Harbor Laboratory (2023-06-26) https://doi.org/gsdsvs
DOI: 10.1101/2023.06.26.546586

26. **The effect of non-linear signal in classification problems using gene expression**
Benjamin J Heil, Jake Crawford, Casey S Greene
PLOS Computational Biology (2023-03-27) https://doi.org/gr2q6q
DOI: 10.1371/journal.pcbi.1010984 · PMID: 36972227 · PMCID: PMC10079219

27. **Rectified linear units improve restricted boltzmann machines**
Vinod Nair, Geoffrey E Hinton
Proceedings of the 27th International Conference on International Conference on Machine Learning (2010-06-21) https://dl.acm.org/doi/10.5555/3104322.3104425
ISBN: 9781605589077

28. **PyTorch: An Imperative Style, High-Performance Deep Learning Library**
Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, … Soumith Chintala

arXiv (2019-12-05) https://arxiv.org/abs/1912.01703

29. **Adam: A Method for Stochastic Optimization**
Diederik P Kingma, Jimmy Ba
arXiv (2017-01-31) https://arxiv.org/abs/1412.6980

30. **Cancer Genome Landscapes**
B Vogelstein, N Papadopoulos, VE Velculescu, S Zhou, LA Diaz, KW Kinzler
Science (2013-03-28) https://doi.org/6rg
DOI: 10.1126/science.1235122 · PMID: 23539594 · PMCID: PMC3749880

31. **EGFR Mutations and Lung Cancer**
Gilda da Cunha Santos, Frances A Shepherd, Ming Sound Tsao
Annual Review of Pathology: Mechanisms of Disease (2011-02-28) https://doi.org/dd359s
DOI: 10.1146/annurev-pathol-011110-130206 · PMID: 20887192

32. **Spectrum of EGFR aberrations and potential clinical implications: insights from integrative pan-cancer analysis**
Haijing Liu, Bo Zhang, Zhifu Sun
Cancer Communications (2020-01) https://doi.org/ghsz4b
DOI: 10.1002/cac2.12005 · PMID: 32067422 · PMCID: PMC7163653

33. **Patient-derived cells from recurrent tumors that model the evolution of IDH-mutant glioma**
Lindsey E Jones, Stephanie Hilz, Matthew R Grimmer, Tali Mazor, Chloé Najac, Joydeep Mukherjee, Andrew McKinney, Tracy Chow, Russell O Pieper, Sabrina M Ronen, … Joseph F Costello
Neuro-Oncology Advances (2020-01-01) https://doi.org/gsfw2p
DOI: 10.1093/noajnl/vdaa088 · PMID: 32904945 · PMCID: PMC7462278

34. **Machine Learning Detects Pan-cancer Ras Pathway Activation in The Cancer Genome Atlas**
Gregory P Way, Francisco Sanchez-Vega, Konnor La, Joshua Armenia, Walid K Chatila, Augustin Luna, Chris Sander, Andrew D Cherniack, Marco Mina, Giovanni Ciriello, … Armaz Mariamidze
Cell Reports (2018-04) https://doi.org/gfspsb
DOI: 10.1016/j.celrep.2018.03.046 · PMID: 29617658 · PMCID: PMC5918694

35. **Predicting Drug Response and Synergy Using a Deep Learning Model of Human Cancer Cells**
Brent M Kuenzi, Jisoo Park, Samson H Fong, Kyle S Sanchez, John Lee, Jason F Kreisberg, Jianzhu Ma, Trey Ideker
Cancer Cell (2020-11) https://doi.org/gh7z2n
DOI: 10.1016/j.ccell.2020.09.014 · PMID: 33096023 · PMCID: PMC7737474

36. **DeepGO: predicting protein functions from sequence and interactions using a deep ontology-aware classifier**
Maxat Kulmanov, Mohammed Asif Khan, Robert Hoehndorf
Bioinformatics (2017-10-03) https://doi.org/gc3nb8
DOI: 10.1093/bioinformatics/btx624 · PMID: 29028931 · PMCID: PMC5860606

37. **Knowledge-primed neural networks enable biologically interpretable deep learning on single-cell sequencing data**
Nikolaus Fortelny, Christoph Bock
Genome Biology (2020-08-03) https://doi.org/gg8ws9
DOI: 10.1186/s13059-020-02100-5 · PMID: 32746932 · PMCID: PMC7397672

38. **Knowledge-guided deep learning models of drug toxicity improve interpretation**
Yun Hao, Joseph D Romano, Jason H Moore
Patterns (2022-09) https://doi.org/gshk7s
DOI: 10.1016/j.patter.2022.100565 · PMID: 36124309 · PMCID: PMC9481960

39. **The Clinical Relevance of Cancer Cell Lines**
J-P Gillet, S Varma, MM Gottesman
JNCI Journal of the National Cancer Institute (2013-02-21) https://doi.org/f4tstr
DOI: 10.1093/jnci/djt007 · PMID: 23434901 · PMCID: PMC3691946

40. **Cancer Cell Lines for Drug Discovery and Development**
Jennifer L Wilding, Walter F Bodmer
Cancer Research (2014-04-30) https://doi.org/f56fwg
DOI: 10.1158/0008-5472.can-13-2971 · PMID: 24717177

41. **A Landscape of Pharmacogenomic Interactions in Cancer**
Francesco Iorio, Theo A Knijnenburg, Daniel J Vis, Graham R Bignell, Michael P Menden, Michael Schubert, Nanne Aben, Emanuel Gonçalves, Syd Barthorpe, Howard Lightfoot, … Mathew J Garnett
Cell (2016-07) https://doi.org/f8wq4s
DOI: 10.1016/j.cell.2016.06.017 · PMID: 27397505 · PMCID: PMC4967469

42. **A pitfall for machine learning methods aiming to predict across cell types**
Jacob Schreiber, Ritambhara Singh, Jeffrey Bilmes, William Stafford Noble
Genome Biology (2020-11-19) https://doi.org/gshk6j
DOI: 10.1186/s13059-020-02177-y · PMID: 33213499 · PMCID: PMC7678316

43. **Navigating the pitfalls of applying machine learning in genomics**
Sean Whalen, Jacob Schreiber, William S Noble, Katherine S Pollard
Nature Reviews Genetics (2021-11-26) https://doi.org/gnm4r9
DOI: 10.1038/s41576-021-00434-9 · PMID: 34837041

44. **HyenaDNA: Long-Range Genomic Sequence Modeling at Single Nucleotide Resolution**
Eric Nguyen, Michael Poli, Marjan Faizi, Armin Thomas, Callum Birch-Sykes, Michael Wornow, Aman Patel, Clayton Rabideau, Stefano Massaroli, Yoshua Bengio, … Chris Ré
arXiv (2023-06-29) https://arxiv.org/abs/2306.15794

45. **scGPT: Towards Building a Foundation Model for Single-Cell Multi-omics Using Generative AI**
Haotian Cui, Chloe Wang, Hassaan Maan, Kuan Pang, Fengning Luo, Bo Wang
Cold Spring Harbor Laboratory (2023-05-01) https://doi.org/gshk6p
DOI: 10.1101/2023.04.30.538439

46. **Large Scale Foundation Model on Single-cell Transcriptomics**
Minsheng Hao, Jing Gong, Xin Zeng, Chiming Liu, Yucheng Guo, Xingyi Cheng, Taifeng Wang, Jianzhu Ma, Le Song, Xuegong Zhang
Cold Spring Harbor Laboratory (2023-05-31) https://doi.org/gshk6q
DOI: 10.1101/2023.05.29.542705

47. **Open collaborative writing with Manubot**
Daniel S Himmelstein, Vincent Rubinetti, David R Slochower, Dongbo Hu, Venkat S Malladi, Casey S Greene, Anthony Gitter
PLOS Computational Biology (2019-06-24) https://doi.org/c7np
DOI: 10.1371/journal.pcbi.1007128 · PMID: 31233491 · PMCID: PMC6611653

Supplementary Material

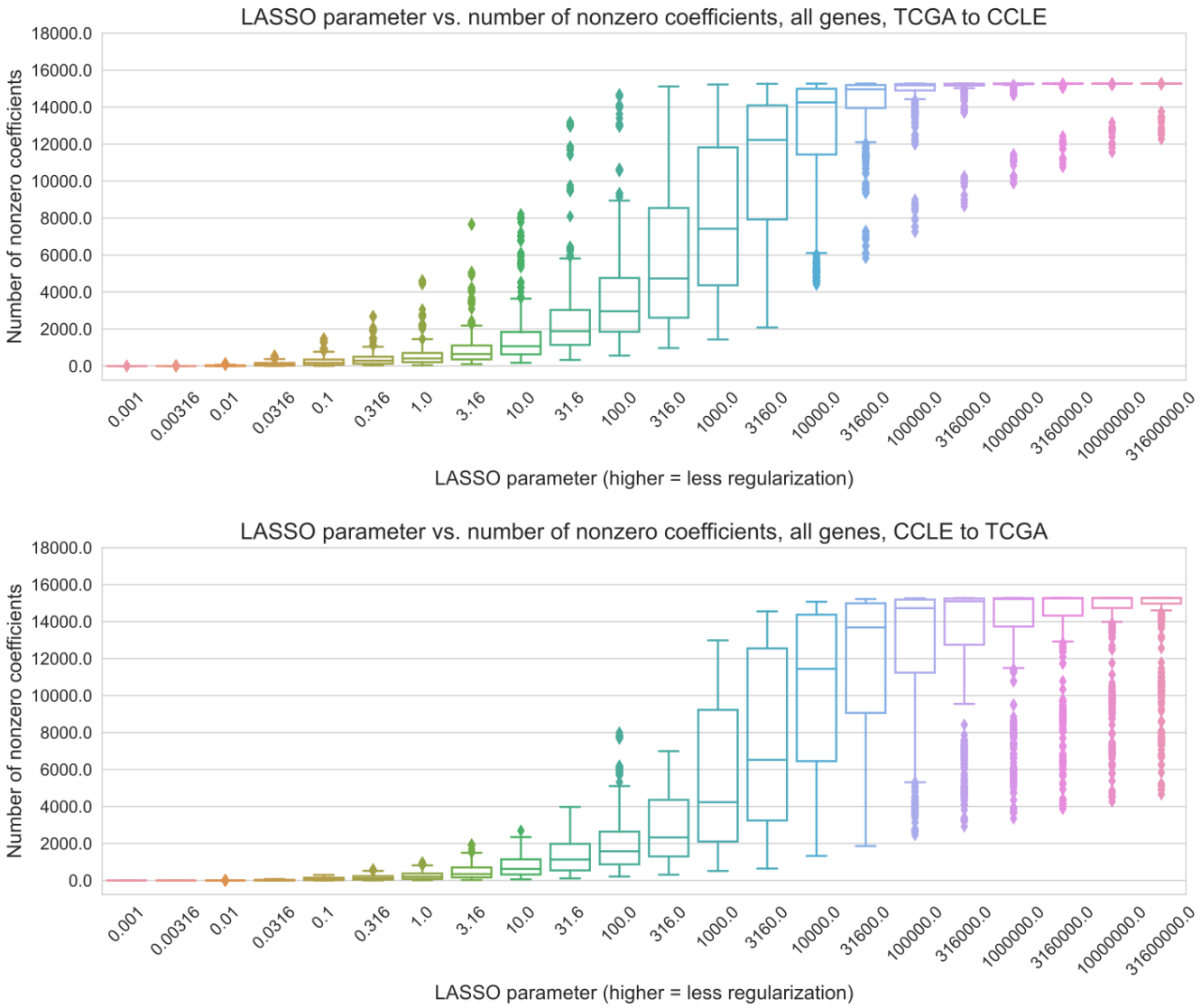

Figure S1: Number of nonzero coefficients (model sparsity) across varying regularization parameters, for 71 genes (TCGA to CCLE prediction, top) and 70 genes (CCLE to TCGA prediction, bottom) in the Vogelstein et al. dataset.

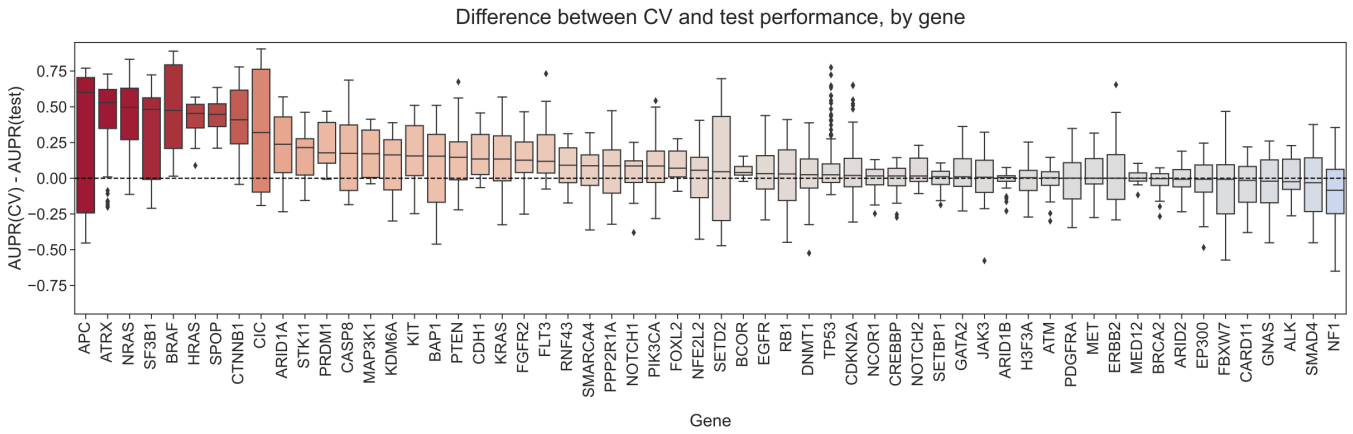

Figure S2: Distributions of performance difference between cross-validation data (same cancer types as training data) and holdout data (cancer types not represented in data), grouped by held-out gene. Each point shows performance for a single train/validation split for one cancer type that was held out, using a classifier trained to predict mutations in the given gene.

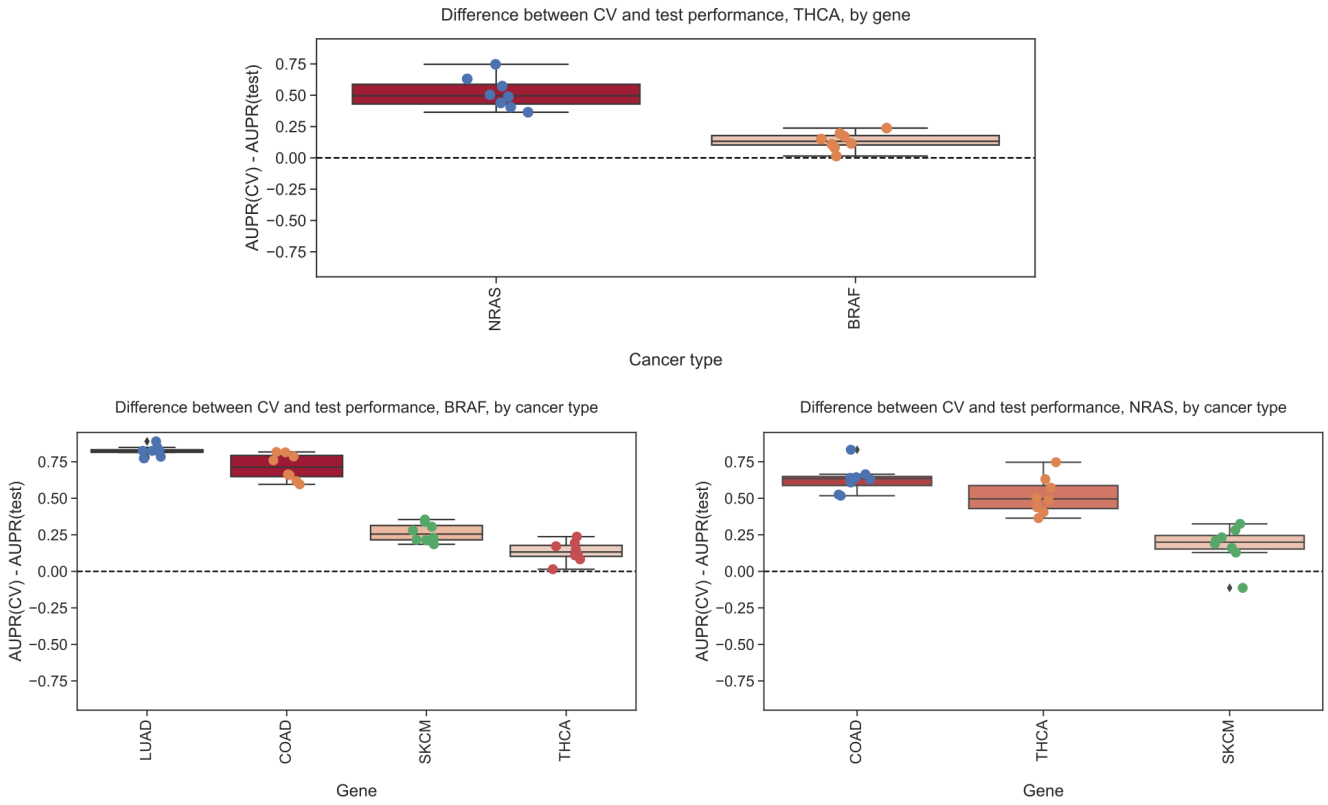

Figure S3: Top row: Distribution of performance differences when thyroid cancer (THCA) data is held out from training setacross seeds/folds, grouped by gene. Bottom row: Distributions of performance differences for genes where THCA is included in training/holdout sets, relative to other cancer types that are included.

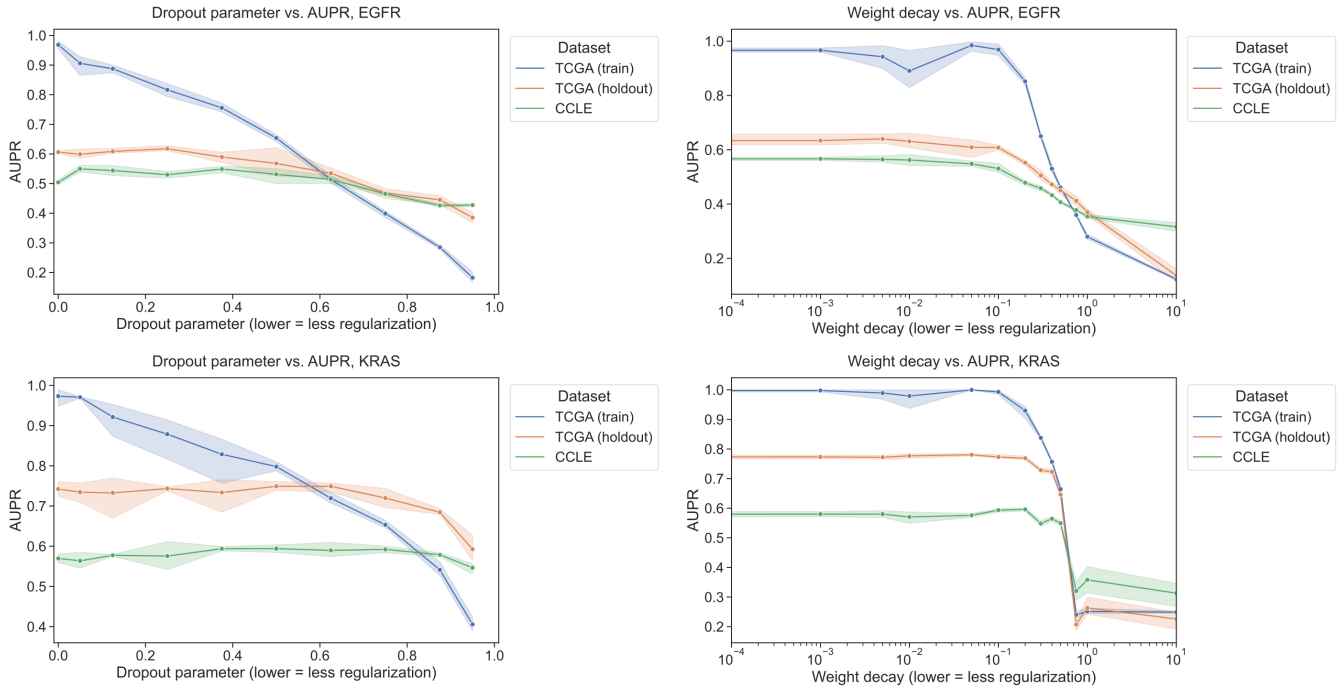

Figure S4: Performance vs. dropout parameter (first column) and weight decay strength (second column), for EGFR mutation prediction (first row) and KRAS mutation prediction (second row) using a 3-layer fully connected neural network trained on TCGA (blue/orange) and evaluated on CCLE (green).